UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/540,219	01/17/2006	Jean-Bernard Fischer	0579-1097	5286

466 7590 03/31/2011

YOUNG & THOMPSON
209 Madison Street
Suite 500
Alexandria, VA 22314

EXAMINER	
VAUGHAN, MICHAEL R	
ART UNIT	PAPER NUMBER
2431	

NOTIFICATION DATE	DELIVERY MODE
03/31/2011	ELECTRONIC

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Notice of the Office communication was sent electronically on above-indicated "Notification Date" to the following e-mail address(es):

DocketingDept@young-thompson.com

UNITED STATES PATENT AND TRADEMARK OFFICE

Commissioner for Patents
United States Patent and Trademark Office
P.O. Box 1450
Alexandria, VA 22313-1450
www.uspto.gov

BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

Application Number: 10/540,219
Filing Date: January 17, 2006
Appellant(s): FISCHER ET AL.

James J. Livingston, Jr.
Reg. No. 55,394
For Appellant

EXAMINER'S ANSWER

This is in response to the appeal brief filed 1/18/11 appealing from the Office action mailed 8/11/10.

(1) Real Party in Interest

The examiner has no comment on the statement, or lack of statement, identifying by name the real party in interest in the brief.

(2) Related Appeals and Interferences

The examiner is not aware of any related appeals, interferences, or judicial proceedings which will directly affect or be directly affected by or have a bearing on the Board's decision in the pending appeal.

(3) Status of Claims

The following is a list of claims that are rejected and pending in the application:

Claims 1-9, 11-21, 23-30, 32, 33, and 35-40 are pending and stand rejected.

(4) Status of Amendments After Final

The examiner has no comment on the appellant's statement of the status of amendments after final rejection contained in the brief.

(5) Summary of Claimed Subject Matter

The examiner has no comment on the summary of claimed subject matter contained in the brief.

(6) Grounds of Rejection to be Reviewed on Appeal

The examiner has no comment on the appellant's statement of the grounds of rejection to be reviewed on appeal. Every ground of rejection set forth in the Office action from which the appeal is taken (as modified by any advisory actions) is being maintained by the examiner except for the grounds of rejection (if any) listed under the subheading "WITHDRAWN REJECTIONS." New grounds of rejection (if any) are provided under the subheading "NEW GROUNDS OF REJECTION."

(7) Claims Appendix

The examiner has no comment on the copy of the appealed claims contained in the Appendix to the appellant's brief.

(8) Evidence Relied Upon

7,168,065	Naccache et al	1-2007

(9) Grounds of Rejection

The following ground(s) of rejection are applicable to the appealed claims:

Claims 1-4, 8, 9, 11-21, 23-28, 32, 33, and 35-40 are rejected under 35 U.S.C. 102(e) as being anticipated by USP 7,168,065 to Naccache et al, hereinafter Naccache.

As per claim 1, Naccache teaches a method of making secure the execution of a computer program (EXE) including a set of at least one instruction, which method is characterized in that it includes:

- a first step (E30), prior to the execution of the computer program, of calculating and storing a first signature (SIG1) representative of the intended execution of the set of instructions (col. 4, lines 25-29),
- a second step (E50), during the execution of the set of instructions, of calculating and storing a second signature (SIG2) representative of the execution of the set of instructions (col. 4, lines 35-36), and
- a step (E60) of detecting an anomaly in the execution of the set of instructions on the basis of the first signature (SIG1) and the second signature (SIG2) (col. 4, lines 38-39), wherein said set of instructions comprising at least one first instruction for initializing the calculation of the second signature (col. 9, lines 25-30 and Fig. 3, element 34), at least one second instruction depending upon the calculation mode of the second signature (Fig. 3, element 40, and col. 9, lines 34-40), and a third instruction, different than the at least one second instruction, for comparing the second signature obtained according to the at least one second instruction with the first signature (col. 9, lines 51-55 and Fig. 3, element 50).

As per claim 26, Naccache teaches a device for making secure the execution of a computer program including a set of instructions comprising at least one instruction,

which device is characterized in that it includes (see abstract):

- a first register (REG1) (col. 4, line 8) for storing a first signature (SIG1) representative of the intended execution of the set of instructions (col. 4, lines 25-29),

- means (22) for calculating and storing in a second storage register (REG2) (col. 6, line 18) during the execution of the set of instructions a second signature (SIG2) representative of the execution of the set of instructions (col. 4, lines 35-36), and

- means (24) for detecting an anomaly in the execution of the set of instructions on the basis of the first signature (SIG1) and the second signature (SIG2) (col. 4, lines 35-36), said set of instructions comprising at least one first instruction for initializing the calculation of the second signature (col. 9, lines 25-30 and Fig. 3, element 34), at least one second instruction depending upon the calculation mode of the second signature (Fig. 3, element 40, and col. 9, lines 34-40), and a third instruction, different than the at least one second instruction, for comparing the second signature obtained according to the at least one second instruction with the first signature (col. 9, lines 51-55 and Fig. 3, element 50).

As per claim 2, Naccache teaches that the first calculation and storage step (E30) is executed during the generation [preparation] of the instructions (AI, AI3) of the computer program (col. 4, line 25).

As per claims 3 and 27, Naccache teaches that the second signature (SIG2) stored during the second calculation and storage step (E50) is retained in memory during the execution of at least one second instruction following the set of instructions

(col. 5, lines 4-6 and 64-68). Naccache teaches using one the preceding values in memory to calculate the next value, so therefore it must remain in memory.

As per claims 4 and 28, Naccache teaches the first signature (SIG1) is obtained from the number of instructions in the set of instructions [accounts for each number of the instructions] (col. 9, lines 23-27),

- the second signature (SIG2) is obtained from the number of instructions from the set of instructions that have been executed [numerical value of executed instructions](col. 9, lines 31-35), and in that

the detection step (E60) detects an execution anomaly when the first signature (SIG1) and the second signature (SIG2) are different after the execution of the set of instructions [compare VHn to Vref] (col. 9, lines 60-64).

As per claims 8 and 32, Naccache teaches the first signature (SIG1) is obtained from the code of a critical instruction of the set of instructions (col. 4, lines 25-29),

- the second signature is obtained from the code of the critical instruction, that code being stored at the same time as or after the execution of the critical instruction [jump] (col. 14, lines 32-35), and in that

- the detection step (E60) detects an execution anomaly when the first signature (SIG1) and the second signature (SIG2) are different after the execution of the set of instructions (col. 10, lines 14-19).

As per claims 9 and 33, Naccache teaches the first signature (SIG1) is obtained from the address of a critical instruction (col. 5, line 51) of the set of instructions, the address being obtained during or after the generation of the executable code of the set

of instructions (col. 4, lines 25-29),

- the second signature (SIG2) is obtained from the address of the critical instruction, that address being stored (E30) at the same time as or after the execution (E30) of the critical instruction (col. 14, lines 32-38), and

- the detection step (E60) detects an execution anomaly when the first signature (SIG1) and the second signature (SIG2) are different after the execution of the set of instructions (col. 10, lines 14-19).

As per claims 11 and 35, Naccache teaches the first signature (SIG1) and the second signature (SIG2) are error detector codes (CRC1, CRC2) calculated from the code or from an address of an instruction of the set of instructions (col. 5, lines 53-58), and in that the detection step (E60) detects an execution anomaly when the first signature (SIG1) and the second signature (SIG2) are different after the execution of the set of instructions (col. 10, lines 14-19).

As per claims 12 and 36, Naccache teaches that the error detector codes are cyclic redundancy check codes (col. 5, lines 53-58).

As per claims 13 and 37, Naccache teaches that the error detector codes are obtained by the logical combination (XOR) of the code or an address of at least one instruction of the set of instructions (col. 5, lines 53-58). Naccache teaches the use of CRC which perform logical combination (XOR included) in order to carry out the operation. Examiner is not giving XOR patentable weight here as the syntax implies XOR as an example of logical combination.

As per claims 14 and 38, Naccache teaches the first signature (SIG1) and the second signature (SIG2) are respectively obtained during the generation and the execution of the instructions from at least two elements chosen from:
the number of instructions in the set of instructions,
the **code** of at least one instruction of the set of instructions (col. 5, lines 45-51),
the **address** of at least one instruction of the set of instructions (col. 5, lines 45-51), and an error detector code calculated from the code or an address of at least one critical instruction of the set of instructions, the address being obtained during or after the generation of the executable code of the set of instructions (col. 5, lines 53-59), and in that the detection step (E60) detects an execution anomaly when the first signature (SIG1) and the second signature (SIG2) are different after the execution of the set of instructions (col. 10, lines 14-19). Naccache teaches using the code and address as hash inputs thus two criteria from the list are chosen.

As per claims 15 and 39, Naccache teaches that it includes a step (E70) of destroying at least a portion of the system on which the computer program is executed, this step of destroying being made when an execution anomaly is detected in the detection step (col. 4, line 45).

As per claim 16, Naccache teaches in that the first signature (SIG1) is generated automatically [already generated before execution of program] (col. 4, line 25-30).

As per claim 17, Naccache teaches a device for processing a computer program including a set of at least one instruction, characterized in that it includes means (12) for

calculating and storing a first signature (SIG1), the first signature (SIG1) stored in a memory and the first signature is representative of the intended execution of the set of instructions prior to the execution thereof (col. 4, lines 25-30), said set of instructions comprising at least one first instruction for initializing the calculation of the second signature (col. 9, lines 25-30 and Fig. 3, element 34), at least one second instruction depending upon the calculation mode of the second signature (Fig. 3, element 40, and col. 9, lines 34-40), and a third instruction, different than the at least one second instruction, for comparing the second signature obtained according to the at least one second instruction with the first signature (col. 9, lines 51-55 and Fig. 3, element 50).

As per claim 18, Naccache teaches the first signature (SIG1) [Vref] are adapted to calculate and store information obtained from the number of instructions of the set of instructions (col. 9, line 65 - col. 10, line 5).

As per claim 19, Naccache teaches the means (12) for calculating and storing the first signature (SIG1) are adapted to obtain and store information obtained from the code of a critical instruction [jump] of the set of instructions (col. 14, lines 33-35).

As per claim 20, Naccache teaches means for generating executable code from the computer program (col. 8, lines 35-36).

As per claim 21, Naccache teaches the means for calculating and storing the first signature (SIG1) are adapted to obtain and store information obtained from the address of a critical instruction (col. 5, line 51), the information being obtained of the set of instructions by the means (14) for generating executable code (col. 8, lines 35-40).

As per claim 23, Naccache teaches that the means (12) for calculating and storing the first signature (SIG1) are adapted to calculate and store information obtained from an error detector code (CRC1) calculated from the code or an address of at least one instruction of the set of instructions (col. 5, lines 53-58).

As per claim 24, Naccache teaches that the error detector code (CRCI) is a cyclic redundancy check code (col. 5, line 57).

As per claim 25, Naccache teaches that the error detector code is obtained by a logical combination (XOR) of the code or an address of at least one instruction of the set of instructions (col. 5, lines 53-58). Naccache teaches the use of CRC which perform logical combination (XOR included) in order to carry out the operation. Examiner is not giving XOR patentable weight here as the syntax implies XOR as an example of logical combination.

As per claim 40, Naccache teaches a microcircuit card [smart card] characterized in that it includes a securing device according to claim 26 (col. 6, lines 27-35).

Claims 5-7, 29, and 30 are rejected under 35 U.S.C. 103(a) as being unpatentable over Naccache.

As per claims 5 and 29, Naccache teaches the first signature (SIG1) is obtained from the number of instructions in the set of instructions [accounts for each number of the instructions] (col. 9, lines 23-27). The calculation performed in these claims is an obvious mathematical variation to those taught by Naccache and in claim 4. Claim 4

calculates a running hash value by each of executed instructions and ultimately compares the final result to the reference hash value. This ensures that each instruction is proper and that the instructions in the set are executed in the correct order. One of ordinary skill in the art could have simply run the hash on the unexecuted instructions and subtracted that value to the reference hash to achieve the same desired result. This provides the same assurance that each proper instruction was executed in the correct order. Once all of the instructions are executed, the value should be zero if they all matched the reference hash value. This is simply an operational design choice. The claim would have been obvious because one of ordinary skill in the art can substitute equivalent known methods which yield predictable results.

As per claims 6 and 30, Naccache teaches that an interrupt of the computer program is triggered when the value of the second signature (SIG2) is below a predetermined threshold (col. 4, lines 40-47).

As per claims 7, Naccache teaches that the first signature (SIG1) and the second signature (SIG2) are retained in memory (col. 1, line 47) during the execution of the program in the same register (REG1) (col. 9, lines 13-17).

(10) Response to Argument

Response to (1)

The arguments presented are substantially the same for each of the independent claims, 1, 17, and 26. They will hereby be addressed together. Appellant argues that the prior art, Naccache fails to teach the three monitoring instructions as claimed. Contrary to Appellant's position, Naccache meets the limitations of the claimed invention. The argued position seems to take a very narrow interpretation of the claims. The speculation as to which invention is more secure is of no concern to the patentability of the claims. The notion that specialized hardware is required is also a moot point. All that the claims require are three instructions. All of these instructions are clearly identifiable in Figure 3 and the accompanying text describing reference Figure 3 that is found at column 10, line 25-column 11, line 63.

The claimed first instruction is for initializing the calculation of the second signature. The first instruction can be mapped to Naccache's first monitoring instruction found at step 32 which then leads to the initialization of creating the running signature (VHn). As the instructions are read and hashed, they are done so by the function F. F is any arbitrary function (col. 5, lines 44-51) which coincides with how the first signature was created. The claimed second instruction does not actually perform a specific instruction per se; rather it is loosely defined as depending upon the calculation mode of the second signature. The Appellant appears to interpret the calculation mode in a narrower view than the claim affords. Nevertheless, Naccache teaches that the

function 'F' is arbitrary and that it could relate to any type of hash function (col. 5, lines 44-51) or CRC function (col. 5, lines 53-57). Not only does Naccache teach multiple types of functions for generating the signatures, but looking at Fig. 3 and 4, it is clear that the calculation mode can be calculated at different times. In Fig. 3, 'F' is performed after reading each instruction of the block of instructions between the first and second monitoring instructions. In the calculation mode of Fig. 4, the function 'F' is not calculated until all instructions between the first and second monitoring instructions have been stored. So not only does Naccache teach multiple ways in which the second signature is calculated, but also multiple types of functions as mentioned above. Both anticipate the broad claim language of "depending upon".

With respect to the claimed third instruction, the claim only requires that it is used for comparing the two signatures. As Naccache teaches, the first and second signatures, Vref and VHn respectively, are compared to one another, which clearly occurs once the second monitoring instruction is observed. This is clearly evidence in figure 3, steps 46 and 50 and column 11, lines 34-60. Thus all of the claim features, including three separate monitoring instructions are clearly taught by Naccache.

Appellant argues that Naccache does not disclose the step of calculating a signature representative of the execution of the set of instructions. The speculation raised as to whether malicious instructions could tamper with the second signature is not meaningful to the discussion of the claimed invention. The claimed invention offers no specific features that even address this potential threat. The claim merely requires a second signature calculated and stored representative of the execution of instructions.

Naccache clearly teaches this same feature (col. 11, lines 26-33) wherein the executed instruction set is hashed and stored in a register.

Response to (2)

Appellant argues the dependent claims are allowable based on their base claims. Per the reasons above and as specified in the Ground of Rejection, dependent claims 5-7, 29, 30 are also unpatentable in view of Naccache.

(11) Related Proceeding(s) Appendix

No decision rendered by a court or the Board is identified by the examiner in the Related Appeals and Interferences section of this examiner's answer.

For the above reasons, it is believed that the rejections should be sustained.

Respectfully submitted,

/MICHAEL R VAUGHAN/

Examiner, Art Unit 2431

Conferees:

/Christopher A. Revak/

Primary Examiner, Art Unit 2431

/William Korzuch/

Supervisory Patent Examiner, Art Unit 2653